

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problems Mailbox.**

**(54) COMMAND INPUT DEVICE FOR WINDOW SYSTEM**

(11) 4-123225 (A) (43) 23.4.1992 (19) JP

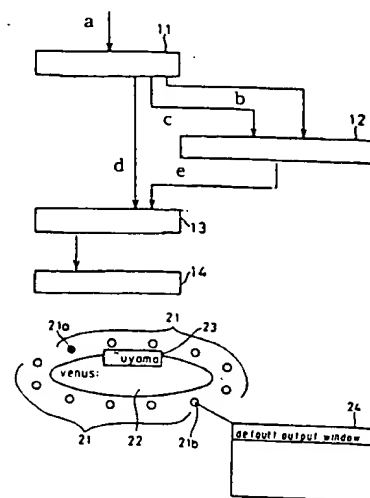
(21) Appl. No. 2-244568 (22) 14.9.1990

(71) FUJITSU LTD (72) MASASHI UYAMA(2)

(51) Int. Cl<sup>5</sup>. G06F3/14

**PURPOSE:** To utilize this command input device in accordance with the skillfulness of each person independently of a beginner or an expert without using a mouse by inputting a command by the combination of an input character string and an active menu item.

**CONSTITUTION:** Twelve satellites 21 in total shown by white and black circles arranged on an elliptic locus are menu items, the white circles 21b are non-active and the black circle 21a is active. When a scanning menu selecting means 12 successively changes active menu items with the lapse of time, a user completes the depression of an input character string at proper timing during the active period of a required menu item. Thereby, the modification, i.e. preprocessing or post processing, of the depressed input character or the change of the character string can be optionally selected based upon the attribute of the required menu item at the time of the completion. Consequently, any person, i.e. a beginner or an expert, can continuously utilize the command input device without using a mouse.



11: buffering device, 13: command modifying device, 14: command interpreting device, a: character input, b: control signal, c: select signal, d: modified character string, e: modifying program

**(54) METHOD FOR EXCHANGING DATA**

(11) 4-123226 (A) (43) 23.4.1992 (19) JP

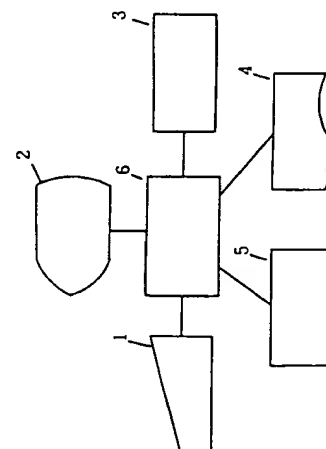
(21) Appl. No. 2-242409 (22) 14.9.1990

(71) HITACHI LTD(1) (72) EIJI HASEGAWA

(51) Int. Cl<sup>5</sup>. G06F5/00

**PURPOSE:** To eliminate the need for the deleting work or the like of data from an unnecessary item after exchanging data by specifying only objective item data being an exchanging item to an item intended to be exchanged and then exchanging data.

**CONSTITUTION:** In a data exchanging function, telephone book data recorded in an IC card are written in the telephone book data reading area of a storage part 3 by an IC card data reading part. Then, data exchanging condition input for updating the contents of a data exchanging condition table in the storage part 3 is executed. Then, a data exchanging direction "address book—telephone book" is judged, and when the decided result is "YES", address book data storing processing is executed to end the data exchanging function. When the decision is "NO", i.e. "address book—telephone book", telephone book data storing processing is executed and then the data of the telephone book data reading area in the storage part 3 are written in the IC card through a reading/writing part 5 to end the data exchanging function. Consequently, the deleting work, etc., of data from an unnecessary time after data exchange is eliminated.



1: input part, 2: display part, 4: printing part, 5: IC card reading part, 6: control part

**(54) PROGRAM CONTROLLER**

(11) 4-123227 (A) (43) 23.4.1992 (19) JP

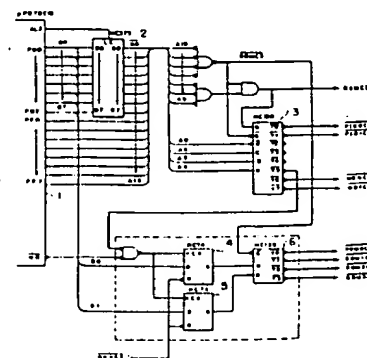
(21) Appl. No. 2-242794 (22) 14.9.1990

(71) FUJI XEROX CO LTD (72) YASUTO KANEKO

(51) Int. Cl<sup>5</sup>. G06F9/06

**PURPOSE:** To freely switch plural ROMs while running a program even after turning on a power supply for a program controller by selecting a ROM to be a storage means to be found out with a storage backing circuit based upon an address signal and a data signal.

**CONSTITUTION:** The storage means backing circuit is constituted of a CPU 1 including address decoders 3, 6 and flip flops (FFs) 4, 5. When data written in specific addresses mapped in a ROM 0 are outputted from data lines D0 to D7, either of HIGH/LOW signals is outputted from the FFs 4, 5 to the A and B ports of the address decoder 3 respectively. A ROM selecting signal is outputted from either of the outputs Y0 to Y3 of the decoder 6 in accordance with the combination of A and B ports HIGH, LOW of the decoder 6. Consequently, the ROMs are switched even in the operation of a machine such as an image processor.



⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A) 平4-123225

⑬ Int. Cl.<sup>9</sup>

識別記号

庁内整理番号

⑭ 公開 平成4年(1992)4月23日

G 06 F 3/14

3 4 0 B

9188-5B

審査請求 未請求 請求項の数 1 (全10頁)

⑮ 発明の名称 ウィンドウシステムのコマンド入力装置

⑯ 特 願 平2-244568

⑰ 出 願 平2(1990)9月14日

⑱ 発 明 者 宇 山 政 志 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内  
⑱ 発 明 者 飯 島 泰 裕 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内  
⑱ 発 明 者 川 口 尚 久 神奈川県川崎市中原区上小田中1015番地 富士通株式会社  
内  
⑲ 出 願 人 富 士 通 株 式 会 社 神奈川県川崎市中原区上小田中1015番地  
⑲ 代 理 人 弁 理 士 伊 東 忠 彦 外2名

明 細 書

1. 発明の名称

ウィンドウシステムのコマンド入力装置

2. 特許請求の範囲

入力文字が制御コマンドのとき制御信号を発行し、該制御コマンドでないときは該入力文字をバッファリングするバッファリング装置(11)と、

常に一つのアクティブなメニュー項目を持ち、時間経過に従って該アクティブなメニュー項目を順次変更する定量的なメニュー選択装置(12)と、

前記バッファリング装置(11)から出力された被修飾文字列と、前記入力文字の入力が完了した時点での前記アクティブなメニュー項目が持つ属性に応じて該定量的なメニュー選択装置(12)から出力された修飾用プログラムとからコマンドを生成するコマンド修飾装置(13)と、

該コマンド修飾装置(13)からのコマンドを解釈実行するコマンド解釈装置(14)と

を有し、同一の入力文字に対する動作を多様に変化させることを特徴とするウィンドウシステムのコマンド入力装置。

3. 発明の詳細な説明

(概要)

ビットマップ表示装置を利用したウィンドウシステムにおける、定量的なメニュー選択装置を利用したコマンド入力装置に関し、

マウスを用いることなく、初心者から熟練者までが連続的に利用できることを目的とし、

入力文字が制御コマンドのとき制御信号を発行し、該制御コマンドでないときは該入力文字をバッファリングするバッファリング装置と、常に一つのアクティブなメニュー項目を持ち、時間経過に従って該アクティブなメニュー項目を順次変更する定量的なメニュー選択装置と、前記バッ

ファリング装置から出力された被修飾文字列と、前記入力文字の入力が完了した時点での前記アクティブなメニュー項目が持つ属性に応じて該走査的なメニュー選択装置から出力された修飾用プログラムとからコマンドを生成するコマンド修飾装置と、該コマンド修飾装置からのコマンドを解釈実行するコマンド解釈装置と、を有し、同一の入力文字に対する動作を多様に変化させるよう構成する。

#### 〔産業上の利用分野〕

本発明はウィンドウシステムのコマンド入力装置に係り、特にビットマップ表示装置を利用したウィンドウシステムにおける走査的なメニュー選択装置を利用したコマンド入力装置に関する。

ビットマップ表示装置とマウスを有し、UNIXオペレーティングシステムを搭載したワークステーションの普及につれて、ウィンドウシステムを利用してソフトウェアの開発が行なわれる機会が多くなってきており、利用しやすいウィンドウシ

ステムが要求されている。このため、いくつかのウィンドウシステムが開発されているが、マウスとアイコンを利用した視覚的なインタフェースでは操作が限定され、UNIXの持つ多様なコマンド体系を利用してソフトウェアを開発するには適当でなかった。従って、UNIXの利点を有したまま、ビットマップ表示装置の利点を活かしたウィンドウシステムのコマンド入力装置が必要とされる。

〔従来の技術〕

第17図は従来の走査的なメニュー選択装置を有するコマンド入力装置の一例の構成図を示す。同図中、セレクト信号が走査的なメニュー選択装置1に入力されるとコマンドを発行してコマンド解釈装置2に渡す。

走査的なメニュー選択装置1は第18図(A)、(B)に示す処理を行なう装置で、同図(A)に示すように、まず初期化を行なう(ステップS1)。この初期化では最初のメニュー項目がア

クティブにされ、時間がゼロにセットされ、かつ、時間限界がユーザ設定値にセットされる。

次に時間は上記時間限界を超えるまで"1"ずつ加算されていき(ステップS2、S3)、時間限界を超えると時間が初期値ゼロにセットされ(ステップS4)、現在アクティブな項目を非アクティブにし、次のメニュー項目をアクティブにした後ステップS2へ戻る(ステップS5)。このようにして、複数のメニュー項目が時間限界で設定された時間毎に順番に一つのみアクティブにされる。

また、メニュー選択装置1は第18図(B)に示すように外部から割り込みによるセレクト信号が入力されると、その時点でアクティブな項目を選択実行し、コマンドを発行する。

ところで、ウィンドウシステムにはマウスによるアイコンの直接操作を基本にしたものと、端末ウィンドウを多用するシステムとの2つの傾向が存在していた。マウスによる直接操作は初心者が利用するには、あるいは定型的な操作を行なうに

は理解し易く都合の良いものである。一方、文字端末環境下で作成されたUNIXのコマンド体系では、利用者はコマンドオプションを指定することで、臨機応変に操作の細かいバリエーションを得ることができ、初心者から熟練者までが習熟度に応じた利用を行なうことができた。後者の端末ウィンドウを多用するシステムはこのUNIXの利点を継承している。

#### 〔発明が解決しようとする課題〕

しかるに、上記の従来のコマンド入力装置では、所望のメニュー項目がアクティブになった時点でセレクト信号を入力するものであり、単純な選択しかできないため、ゲーム機や、障害者のコミュニケーション補助装置のように、入力インタフェースが限定された場合にしか利用されていなかった。

また、マウスによる直接操作を基本にしたウィンドウシステムでは、熟練者が状況に応じた操作の細かいバリエーションを利用することが困難で

あり、一方、端末ウィンドウを多用するシステムでは入力が見覚的な位置関係を利用したものではないため、ビットマップ表示装置の利点を有効に利用していない。

更に前記したウィンドウシステムはいずれもマウス・キーボード間の手の移動が煩雑に行なわれ、手の移動によって利用者の思考が中断されるという問題を生じていた。

本発明は上記の点に着目してなされたもので、マウスを用いることなく、初心者から熟練者までが連続的に利用できるウィンドウシステムのコマンド入力装置を提供することを目的とする。

#### 〔課題を解決するための手段〕

第1図は本発明の原理説明図を示す。同図中、11はバッファリング装置で、入力文字が制御コマンドのとき制御信号を発行し、制御コマンドでないときは入力文字をバッファリングする。

12は走査的なメニュー選択装置で、常に一つのアクティブなメニュー項目を持ち、時間経過に

ることが可能になり、初心者と熟練者のコマンドの使い分けや、マウスを用いずにウィンドウやメニューを選択することが可能になる。

#### 〔実施例〕

第2図は本発明になるウィンドウシステムのコマンド入力装置におけるビットマップ表示装置上での表示態様の一実施例を示す。同図中、楕円軌跡上に配置された計12個の白丸及び黒丸21はメニュー項目で、本実施例ではそれを表示形状から「衛星」と呼ぶものとする。衛星21は白丸が非アクティブで、黒丸21aがアクティブである。衛星21はウィンドウあるいはアイコンを属性として持ち、ウィンドウあるいはアイコンは、プロセス、ファイル、ディレクトリのうち何れかの種類である。アクティブな衛星21aは、設定した表示色で表示され、通常は右回りに所定時間毎に推移していく。

また、衛星21により取り囲まれている楕円はエコーバック領域22で、ユーザの文字列入力を

従ってアクティブなメニュー項目を順次変更する。13はコマンド修飾装置で、バッファリング装置11から出力された被修飾文字列と、入力文字の入力が完了した時点でのアクティブなメニュー項目が持つ属性に応じて走査的なメニュー選択装置12から出力された修飾用プログラムとからコマンドを作成する。更に、14はコマンド解釈装置で、コマンド修飾装置13からのコマンドを解釈実行する。

#### 〔作用〕

本発明では、第1図における走査的なメニュー選択装置12が時間経過に従ってアクティブなメニュー項目を順次変更している状態において、利用者がタイミングを見計らって所望のメニュー項目がアクティブになっている時に入力文字列の打鍵入力を完了することで、その時点での所望のメニュー項目の属性により、打鍵した入力文字列の修飾、つまり前処理や後処理を加えたり、文字列自体に変更を加えたりすることを、任意に選択す

そのまま表示する。23は現在のディレクトリの表示部である。更に24はデフォルト出力ウィンドウで、衛星21bの属性である。

このようなビットマップ表示装置上での表示を見ながら、ユーザは所望のコマンドをタイミングを見計らってキーボードから入力する。本発明のコマンド入力装置は第1図に示したように、バッファリング装置11、走査的なメニュー選択装置12、コマンド修飾装置13、コマンド解釈装置14からなり、次にこれらの各装置11～14の実施例について夫々説明する。

第3図はバッファリング装置の一実施例の構成図を示す。同図中、バッファリング装置11は、256バイト文字バッファ111と、制御コマンドテーブル112とを有し、文字入力があったときに動作を開始し、まず第4図に示す制御テーブル112を参照する。ここで、制御テーブル112は第4図に示すように、制御コマンドを示す文字とその制御信号とが対応付けられたテーブルで、MとDは夫々セレクト、Sは前記アクティブな

衛星218の移動を停止するためのストップ、  
・Fはアクティブな衛星218を右回りに回転させるフォワード、  
・Bは左回りに回転させるバック、  
・Rはアクティブな衛星218の回転方向を反転するリバース、  
・Qはストップにより停止した衛星の移動を再開するムーブの各制御コマンドである。

かかるバッファリング装置11は第5図のフローチャートに従った処理動作を行なう。すなわち、文字入力があると、制御コマンドテーブル112を参照して入力文字がセレクトコマンドか否か判定し(ステップ51)、セレクトコマンドのときは文字バッファ111にその入力文字を追加し(ステップ52)、セレクト信号を走査的なメニュー選択装置12に発行した後(ステップ53)、文字バッファ111の内容をコマンド修飾装置13に対する被修飾文字列として書き出す(ステップ54)。

一方、ステップ51で入力文字がセレクトコマンドでないと判定されたときは、他の制御コマン

ドかどうか再び制御コマンドテーブル112を照して調べ(ステップ55)、の制御コマンドのときは該当する制御信号を走査的なメニュー選択装置12に対して発行する(ステップ56)。また入力文字が制御コマンドに該当しない通常の文字の場合は、単純に文字バッファ111に追加される(ステップ57)。

次に走査的なメニュー選択装置12の一実施例について説明する。第6図は走査的なメニュー選択装置12の構造図を示す。メニュー項目に相当するのが前記した衛星21であり、その状態及び前状態は回転、停止、実行中のいずれかであり、その回転方向は右又は左である。

第7図は走査的なメニュー選択装置12の通常の動作時の一実施例の処理の流れを示す。同図中、まず初期化により状態が回転とされ、アクティブな衛星の色を白とし、速度を100、方向を右、時間をゼロ、時間限界を300に夫々設定する(ステップ71)。次に、状態が回転でなければ、そのまま待機し、回転であれば(ステップ72)、

時間が時間限界を超えるまで1ずつ増加され(ステップ73、74)、時間が時間限界を超えると時間を初期値ゼロにセットし(ステップ75)、その後今までアクティブな衛星の表示を白とし、方向が右のときはアクティブとする衛星を次(右側の)衛星とし、方向が左のときはアクティブとする衛星を前(左側の)衛星とし、今回アクティブとする衛星の色を表示色として指定された色とし(ステップ76)、ステップ72へ戻る。

このようにして、ビットマップ表示装置上の衛星は第2図と共に説明したように、このメニュー選択装置12によりアクティブな衛星が所定時間毎に所望の方向の隣接する衛星へ所定の表示色で順次移動する。

ここで、第7図は通常の衛星が回転するときの走査的なメニュー選択装置の処理であるが、割り込み信号に対しては第8図に示す如く処理を行なう。割り込み信号は前記したセレクト信号あるいは制御信号のことで、制御コマンド「set p」、「move」のときの制御信号入力時は第8図

(A)、(B)に夫々示す如く、状態を「停止」、「回転」とする。また、「forward」のときの制御信号入力時は第8図(C)に示す如く状態が停止であるときのみ有効で、それまでアクティブな衛星の表示色を白とし、方向が右(左)のときはアクティブとする衛星を右(左)隣の衛星とし、その表示色を所定の色とする。

「back」の制御信号入力時は第8図(D)に示す如く、「forward」の制御信号入力時と反対方向への移動処理を行なう。また、「reverse」の制御信号入力時は第8図(E)に示すように方向が現在の方向と反対方向になるよう処理する。

また、セレクト信号入力時は第8図(F)に示す如く、現在の状態を保存した上でリターン以外のすべての割り込みを禁止し、アクティブな衛星の表示色を非アクティブなものと同じ白とし、アクティブな衛星へのポイントをコマンド修飾装置13へ送る。また、実行中の走査的なメニュー選択装置12は、「return」の制御

号(リターン番号)により、第8図(G)に示す如くアクティブな衛星の色をアクティブであることを示す表示色にし、かつ、割り込み禁止解除を行なって元の状態に復帰する。

次に衛星の構成及び配置について説明する。第9図に示す衛星の一実施例の構成において、衛星は前記したようにメニュー項目に相当し、属性としてアイコン又はウィンドウを持ち、種類としてプロセス、ファイル、ディレクトリの何れかを持つ。また、衛星は第10図に示す如く、第2図と共に説明したようにビットマップ表示装置の情内軌跡上に12個配置され、そのうち26はデフォルト衛星、27はダミー衛星、28はビジュアル衛星で、デフォルト衛星26がアクティブであるとき、コマンド実行結果の出力がデフォルトウィンドウに表示され、またビジュアル衛星28がアクティブであるとき、コマンド実行結果の出力はメニュー形式で表示されるように修飾される。また、上記の3種類の衛星26~28の夫々は非アクティブのときは同じ白色で表示され

るが、アクティブのときは互いに異なる色で表示されるため、衛星の種類識別が可能となる。

第11図は衛星の持つ修飾用プログラムテーブルの構造を示し、修飾用プログラムテーブルは被修飾文字列のカテゴリと属性又は種類と修飾用プログラムとからなる。

この修飾用プログラムテーブルの構成を第10図の各衛星の場合を例にとりて更に詳細に示すと第12図に示す如くなる。第12図(A)はデフォルト衛星26の修飾用プログラムテーブル、同図(B)はダミー衛星27の修飾用プログラムテーブル、同図(C)はビジュアル衛星28の修飾用プログラムテーブルを示す。これらの修飾用プログラムテーブルはコマンド修飾装置13が利用する。

次にコマンド修飾装置13について説明する。コマンド修飾装置13は第13図に示す如く被修飾文字列と修飾用プログラムとを受け取ることで動作を開始し、まず被修飾文字列のカテゴリをコマンドカテゴリテーブル130に基づき決定する。

コマンドカテゴリテーブル130は第14図に示す如く、カテゴリと文字列とを対応させた表で、例えば文字列が「vi」のときはカテゴリは「window-create」と決定する。

更に、コマンド修飾装置13は第15図に示すフローチャートに従ってコマンドを生成して出力する。すなわち、コマンド修飾装置13は被修飾文字列をコマンドカテゴリテーブル130に基づき分類し(カテゴリを決定し)(ステップ151)、次いでアクティブな衛星の修飾プログラムテーブルを参照し、その中で上記のカテゴリと衛星の属性、種類が一致する項を探す(ステップ152)。

次に最初に一致した項の修飾プログラム中の変数\$commandを被修飾文字列で置き換え(ステップ153)、修飾プログラム中の他の\$process、\$window等の変数を、該当する属性で置き換え(ステップ154)、このように置換されたプログラムをコマンドとしてコマンド解釈装置14へ出力する(ステップ155)。

このようにして、例えばアクティブな衛星が第

16図(A)に示す如く「vi e D」と入力された時点でデフォルト衛星26aであるときは、バッファリング装置11が走査的なメニュー選択装置12へセレクト信号を発行すると共に「vi e D」なる文字列を被修飾文字列としてコマンド修飾装置13へ発行し、このときの入力文字列のカテゴリがlistingなので、「vi e D」がまず解釈され(ここでは最後の「D」がeで始まるファイルをリスト表示する)、出力がデフォルトウィンドウ24に表示される。

また、上記の入力文字列が第16図(B)に示す如くビジュアル衛星28aがアクティブな時点で入力されたものとする、と、「vi e D」の出力はフィルタmake Menuで同図に30で示す如くメニュー化される。

本実施例ではコマンド入力を所望のアクティブな衛星の表示タイミングに合わせて入力することでメニューを選択しコマンドを出力するので、初心者でも使い易く、また利用者が状況に応じて修飾プログラムなどについて多様な変更を加える

ことができるため、熟練者も利用でき、更にマウスを極力使用しないようにできるので、手の移動によって利用者の思考が中断されることも殆どない。

(発明の効果)

上述の如く、本発明によれば、入力文字列とアクティブとするメニュー項目との組み合わせによりコマンドを入力するようにしているため、初心者から熟練者まで習熟度に応じた利用ができ、またマウスの利用を極力避けることができ、以上よりウィンドウシステムを用いたソフトウェア開発の生産性向上に寄与するところ大である等の長を有するものである。

4. 図面の簡単な説明

第1図は本発明の原理説明図、

第2図は本発明におけるビットマップ表示装置上での表示態様の一実施例を示す図、

第3図はバッファリング装置の一実施例の構成

図の構成図、

第15図はコマンド修飾装置の一実施例の処理の流れを示す図、

第16図は本発明の一実施例の表示例を示す図、

第17図は従来装置の一例の構成図、

第18図は走査的なメニュー選択装置の処理の流れを示す図である。

図において、

11はバッファリング装置、

12は走査的なメニュー選択装置、

13はコマンド修飾装置、

14はコマンド解釈装置、

21は衛星、

21aはアクティブな衛星、

22はエコーバック領域を示す。

図、

第4図は第3図の制御コマンドテーブルの構成図、

第5図はバッファリング装置の一実施例の処理の流れを示す図、

第6図は走査的なメニュー選択装置の一実施例の構成図、

第7図は走査的なメニュー選択装置の一実施例の処理の流れを示す図、

第8図は走査的なメニュー選択装置の割り込み信号に対する処理の流れを示す図、

第9図は衛星の一実施例の構成を示す図、

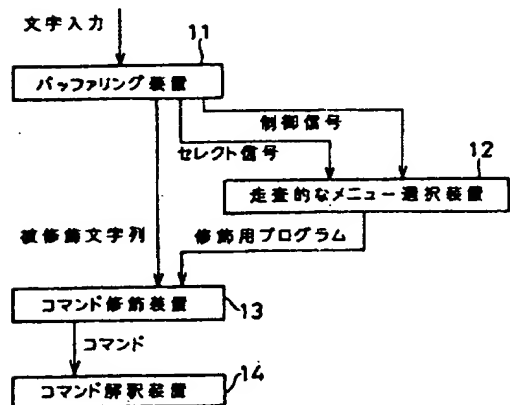
第10図は衛星の配置を示す図、

第11図は修飾プログラムテーブルの構成を示す図、

第12図は第10図の各衛星の修飾プログラムテーブルの構成を示す図、

第13図はコマンド修飾装置の一実施例の構成図、

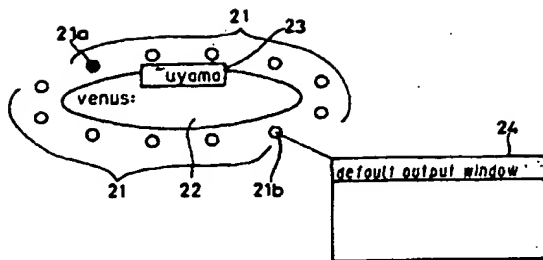
第14図は第13図のコマンドカテゴリテーブ



本発明の原理説明図

第1図





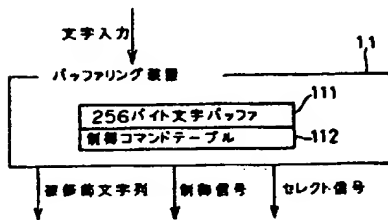
本発明におけるビットマップ表示装置上での表示態様の一例図

第 2 図

走査的なメニュー選択装置の構造	
状態	(回転or停止or実行中)
前状態	(回転or停止or実行中)
方向	(右or左)
時間限界	数値値
時間	数値値
アクティブ	衛星へのポインタ
衛星[12]	衛星の配列

走査的なメニュー選択装置の一実施例の構造図

第 6 図



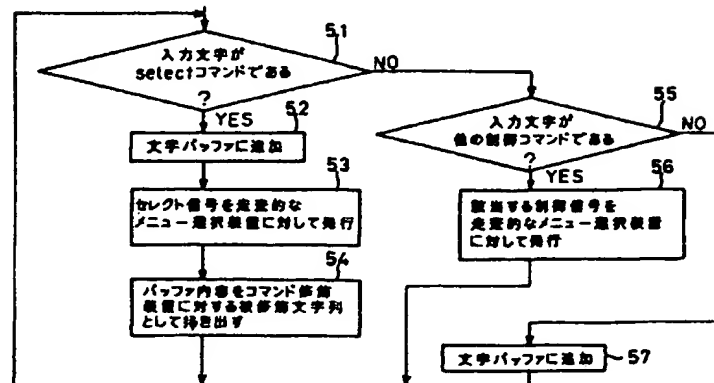
バッファリング装置の一実施例の構成図

第 3 図

制御コマンドテーブル	
文字	制御信号
'M'	select
'D'	select
'S'	stop
'F'	forward
'B'	back
'R'	reverse
'O'	move

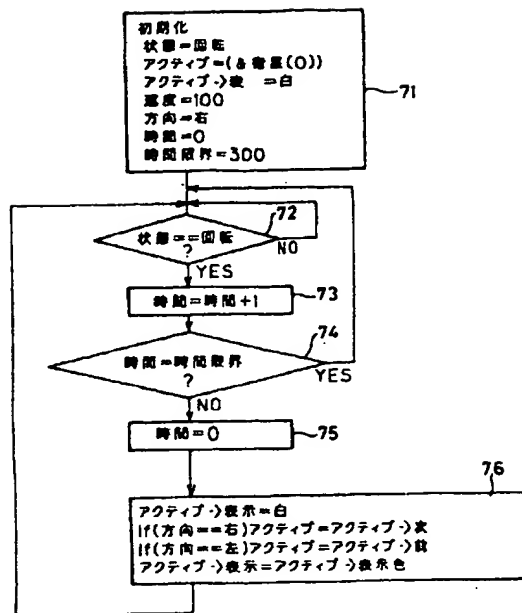
第3図の制御コマンドテーブルの構成

第 4 図



バッファリング装置の一実施例の処理の流れ

第 5 図



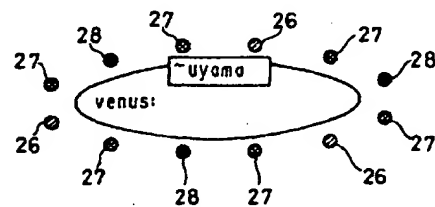
走査的なメニュー選択装置の一実施例の処理の流れ

第 7 圖

衛星(メニュー)の構成	
属性	アイコンorウィンドウorなし
種類	プロセスorファイルorディレクトリorなし
表示色	色
表示	色
前文	衛星へのポインタ
window	衛星へのポインタ
icon	ウィンドウへのポインタ
process	アイコンへのポインタ
file	プロセス番号
directory	ファイル名
position	ディレクトリ
テーブル	座標値
	修飾プログラムテーブルへのポインタ

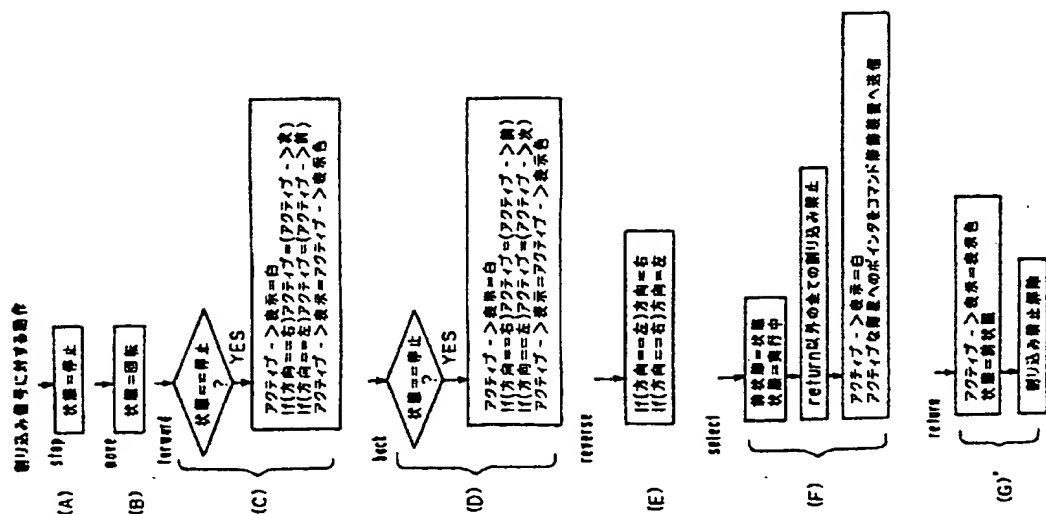
衛星の一実施例の構成図

第 9 図



## 衛星の配置

第 10 圖



### 走査的なメニュー選択機構の創り込み等に対する処理の流れ

88

修飾用プログラムテーブルの構成			
被修飾文字列	カテゴリ	属性or種類	修飾用プログラム

修飾プログラムテーブルの構成

第 11 図

デフォルト荷量の修飾用プログラムテーブル		
(A)	for-window	always return
	for-icon	always return
	for-process	always \$command) \$default-window-ty
	window-create	always return
	text-create	always \$command) \$default-window-ty
	listing	always \$command) \$default-window-ty
	simple	always \$command) \$default-window-ty

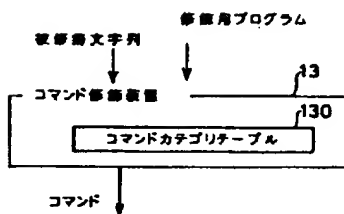
ダミー荷量の修飾用プログラムテーブル		
(B)	always	always stop

ビジュアライズ荷量の修飾用プログラムテーブル		
(C)	for-window	window \$command \$window
	for-icon	icon \$command \$icon
	for-process	process \$command \$process
	window-create	none jterm-e \$command
	text-create	none \$command) temp; jterm-e less \$temp
	listing	none \$command) makeMenu
	simple	always \$command) \$default-window-ty

第10図の各荷量の修飾プログラムテーブルの構成

第 12 図



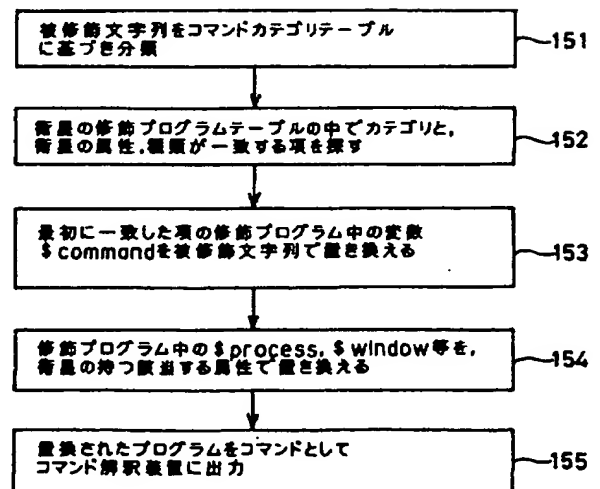
コマンド修飾装置の一実施例の構成図

第 13 図

コマンドカテゴリテーブル	
カテゴリ	文字列
for-window	(iconify,move,resize,...)
for-icon	(open,...)
for-process	(kill,fg,bg,...)
window-create	(vi,emacs,/bin/sh,dbx,ra,...)
text-create	(make,cc,cat,...)
listing	
simple	

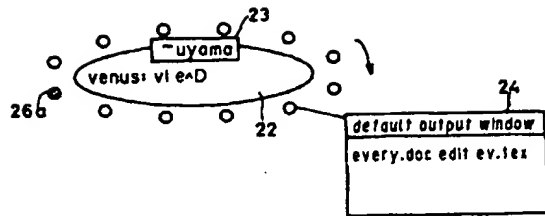
第13図のコマンドカテゴリテーブルの構成図

第 14 図

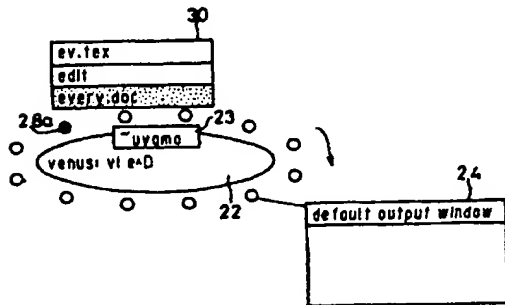


コマンド修飾装置の一実施例の処理の流れ

第 15 図



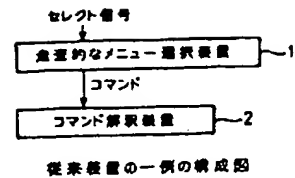
(A) デフォルトウィンドウへの出力



(B) メニュー形式の出力

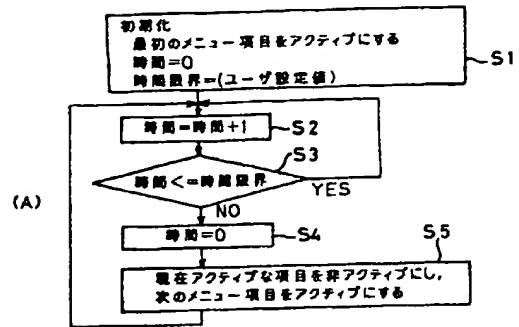
本発明の一実施例の表示例

第 16 図



従来装置の一例の構成図

第 17 図



(A)

割り込みによるセレクト信号

(B)



変動的なメニュー選択装置の処理の流れ  
第 18 図